Licensed Material — Property of IBM
LY20-2311-0

VM 17
new

Installed User Program

Display Editing System for CMS System Guide

Program Number: 5796-PJP
Feature Number: 8654

This manual provides detailed descriptions of the modules which comprise the editing components of the Display Editing System Installed User Program.

The Display Editing System Installed User Program is a full-screen editing system for the Virtual Machine Facility/370 Conversational Monitor System (VM/370 CMS) designed to utilize the features of both local and remote terminals. In addition to a comprehensive array of cursor and command oriented data manipulation functions, the IUP has direct usability in a wide range of user applications. Display terminals supported are the IBM 3277 Model 2 and the IBM 3278 Model 2, including both the TEXT PROCESSING and APL keyboards.

The Display Editing System accommodates CMS files of all types, including program source material, documentation, and data. It permit users to change portions of the display without program intervention. By placing the cursor on the position(s) requiring change, the user may re-enter the correct character(s). The Display Editing System checks for modifications made to the display image and makes permanent copies of the changes to the file.

IBM

PROGRAM SERVICES

Central Service will be provided until otherwise notified. Users will be given a minimum of six months notice prior to the discontinuance of Central Service.

During the Central Service period, IBM through the program sponsor(s) will, without additional charge, respond to an error in the current unaltered release of the program by issuing known error correction information to the customer reporting the problem and/or issuing corrected code or notice of availability of corrected code. However, IBM does not guarantee service results or represent or warrant that all errors will be corrected.

Any on-site program service or assistance will be provided at a charge.

WARRANTY

EACH LICENSED PROGRAM IS DISTRIBUTED ON AN 'AS IS' BASIS WITHOUT WARRANTY OF ANY KIND EITHER EXPRESS OR IMPLIED.

First Edition (September 1977)

A form for readers' comments has been provided at the back of this publication. If this form has been removed, address comments to: IBM Corporation, 1133 Westchester Avenue, White Plains, New York 10604. Attention: R. F. Walsh.

© Copyright International Business Machines Corporation 1977

LICENSED MATERIAL – PROPERTY OF IBM

TABLE OF CONTENTS

INTRODUCTION.....1
MODULE DESCRIPTIONS 2
EDIT0 2
EDITDSP - Display Processor 2
EDPPRC - Post Processing 3
EDITIII - Type III 4
EDITFID - Fileid 4
EDIT1 5
EDCALL - "Call" routines 5
EDSCAN - "Parce" routines 5
EDCVN - Conversion routines 7
EDSTORE - Storage management 9
EDFNCADD - Add records 10
EDFNCDEL - Delete records 11
EDFNCFRD - Read a file into storage 12
EDFNCWR - Write a file to disk 13
EDFNCSTT - STATE a file 15
EDPNT - "Point" name processing 16
EDIT2 18
EDUPD - "Update" facility 18
EDUPDI - Update a base file 19
EDUPDO - Create an update file 21
EDIT3 24
EDLVL - Establish a new edit level 25
EDPROC - Process EDGAR $PROFILE 26
EDLBD - Establish logical display 27
EDGAR - Initial entry and termination 28
EDIO 31
EDIOA - SOS facility 32
EDIOB - Physical display routine 33
EDSVC - SVC X'ED' 34
EDCMD 36
EDCMD0 Process Type I & II commands 36
EDCMD1 Misc. Type I & II commands 38
EDCMD2 Misc. Type I & II commands 38
EDCMD3 Misc. Type I & II commands 39
EDIMSG 40
EDIMSG - Error messages 40

LICENSED MATERIAL - PROPERTY OF IBM

CONTROL BLOCKS 41
EDISECT 41
LVLSECT 42
LDBSECT 44
SAVEAREA 46
IOSECT 47
REC 49
STORE 49
PTSECT 49
PROSECT 50
CRT1 51
CRT2 52
CRT3 52
Equates 53

ECOMMAND 54

LICENSED MATERIAL - PROPERTY OF IBM

I. INTRODUCTION

The DISPLAY EDITING SYSTEM is a general purpose data file editing system for CMS data files of all types. It was developed under VM/370 CMS Release 3 and was designed to use the CMS 3277 console, in "full-screen" mode, as it's communication medium. The "full-screen" operation of a CMS 3277 console requires the installation of IBM PRPQ 5799-AWP, "Extended Full-Screen 3270 Console Interface for VM/370". Also, if remote display devices will be used, the 3271 Control Units must be equipped with EC's 717956, 739540, 739278, and the fix to VM/370 Release 3 apar VM07289 must be installed.

The DISPLAY EDITING SYSTEM is comprised of several components.

- EDGAR is the basic editing system. It is invoked as a CMS command and may be executed in either the user program area (x'20000') or in a Discontiguous Shared Segment. EDGAR's primary functions are to edit a virtual storage copy of a data file, and to manage the display terminal.

- $PROFILE is the filetype of a user modifiable set of specifications which establishes the editing environment.

- Preprocessor EXEC's are automatically invoked, based on filetype, to establish default editing parameters.

- ECOMMAND is a CMS command which interfaces directly between EXEC's and the editing functions of EDGAR.

The EDGAR component of the DISPLAY EDITING SYSTEM is composed of the following modules. They are described in greater detail on the following pages.

EDIT0 - The Display Processor
EDIT1 - The Editing Command Pocessor support functions
EDIT2 - The Update facility
EDIT3 - The Editing Command Processor
EDIO - The I/O Processor
EDCMD - The Editing Commands
EDIMSG - The messages

LICENSED MATERIAL - PROPERTY OF IBM

EDITQ

FUNCTION -

THE MAJOR ROUTINES (CSECTS) IN THIS SOURCE MODULE COMPRISE THE DISPLAY PROCESSOR COMPONENT OF THE SYSTEM.

ENTRY EDITDSP

FUNCTION -

CENTRAL DISPLAY ROUTINE - BUILDS A FORMATTED DISPLAY ON THE SCREEN ACCORDING TO CONTROL BLOCK INFORMATION

ATTRIBUTES -

RE-ENTRANT; ENTERED VIA 'CALL' MACRO

ENTRY CONDITIONS -

NONE

EXIT CONDITIONS -

NONE

REGISTER USAGE -

R11 - EDISECT
R10 - LVLSECT
R9 - LDBSECT
R8 - DDB WIDTH (CONSTANT)
R7 - LAST 'LDBREC' ENTRY ADDRESS
R5 - CURRENT 'LDBREC' POINTER
R4 - DDB POINTER
R3 - ATTRIBUTE BYTE ADDRESS FOR CURRENT DDB

OPERATION -

1. SCAN LDB'S. IF NON-SEQUENTIAL DISPLAY SET-UP BY COMMAND PROCESSOR, LDBREC IS COMPLETE - GO TO 2; ELSE, FORMAT 'LDBREC' IN STANDARD

LICENSED MATERIAL - PROPERTY OF IBM

SEQUENTIAL FASHION.

2. USING INFORMATION IN 'LDBREC', FILL-IN EACH DATA-DISPLAY BUFFER WITH THE "VIEWED" COLUMNS OF THE CORRESPONDING STORAGE RECORD, OR CONSTANTS INDICATING 'TOF', 'EOF', OR 'DEL'.
3. COMPLETE INFORMATION IN TYPE III COMMAND AREA IF PRESENT; IF NONE, GO TO STEP 4.
4. REPEATS STEPS 1, 2, AND 3, FOR EACH LOGICAL DISPLAY. THEN COMPLETE THE CRT STATUS AREA AND EXIT.

ENTRY EDPPRC

FUNCTION -

TO PERFORM THE POST-PROCESSING REQUIRED BY TYPE I COMMANDS ISSUED ON THE PREVIOUSLY PROCESSED EDIT LEVEL.

CALLED BY -

'EDITPRC' (CENTRAL EDIT LEVEL PROCESSOR) AFTER ALL PROCESSING HAS BEEN COMPLETED ON A GIVEN EDIT LEVEL.

'EDLVL' (EDIT LEVEL INITIALIZATION ROUTINE) IF A 'DEFAULT' (PRE-PROCESSOR) EXEC HAS BEEN CALLED.

ENTRY CONDITIONS -

R15 - MODULE ENTRY POINT ADDRESS
R13 - STANDARD REGISTER SAVE AREA ADDRESS
R11 - 'EDISECT' BASE REGISTER
R10 - 'LVLSECT' BASE REGISTER (FOR LEVEL ON WHICH TO PERFORM POST-PROCESSING)

OPERATION -

POST-PROCESSING FUNCTIONS TO BE PERFORMED ARE INDICATED IN 'LVLSW3' (POST-PROCESSING STATS). EACH FUNCTION REQUESTED IS PERFORMED IN SEQUENCE. ERROR MESSAGES ARE PRESENTED IN CONSOLE DISPLAY MODE. ANY ERROR TERMINATES THE POST-PROCESSING AND IT IS THE RESPONSIBILITY OF THE USER TO RE-ISSUE ANY TYPE I COMMANDS, AS REQUIRED TO RETRY THE DESIRED POST-PROCESSING.

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDITIII

FUNCTION -

THIS CONTROL SECTION IS CALLED TO PROCESS IMMEDIATE (TYPE III) COMMANDS.

ENTRY CONDITIONS

R15 = ENTRY POINT ADDRESS
R14 = RETURN ADDRESS
R13 = STD. SAVE AREA ADDRESS
R11 = 'EDISECT' BASE
R10 = 'LVLSECT' BASE
R9 = 'LDBSECT' BASE FOR LOGICAL DSPLY. BEING PROCESSED.
R1 = ADDRESS OF THE TYPE III AREA TO BE PROCESSED ('CRT3' BASE).
R0 = ADDRESS OF 'REC' CORRESPONDING TO THE TYPE III AREA BEING PROCESSED.

ENTRY EDITFID

FUNCTION -

TO PROCESS A CHANGE IN LEVEL FILEID AS A RESULT OF CHANGES TO THE FILEID PORTION OF 'CRT1'

ENTRY CONDITIONS -

R15 - CSECT ENTRY POINT ADDRESS
R13 - STANDARD SAVE AREA BASE REG.
R11 - 'EDISECT' BASE REG.
R10 - 'LVLSECT' BASE REG.

LICENSED MATERIAL - PROPERTY OF IBM

FUNCTION -

THIS SOURCE MODULE CONTAINS MINOR SUPPORT FUNCTIONS FOR THE EDIT PROCESSOR COMPONENT. THESE ROUTINES PERFORM THE BASIC FUNCTIONS REQUIRED BY OTHER EDIT SYSTEM ROUTINES.

ENTRY EDCALL

FUNCTION -

THIS CONTROL SECTION SUPPORTS THE INTER-MODULAR 'CALL' MACRO USED WITHIN THE EDITOR AND IS RESPONSIBLE FOR ASSURING CLEAN AND RAPID SAVE AREA HANDLING DURING THE EDITING PROCESS.

ENTRY EDSCAN

FUNCTION -

TO 'PARSE' REMAINING DATA FOR NEXT ARGUMENT

ENTRY CONDITIONS -

R1 POINTS TO 2 FULL-WORD PARAMETER LIST CONTAINING:
1. ADDRESS OF DATA TO BE PARSED
2. LENGTH (REMAINING) OF DATA TO BE PARSED
R11 STANDARD BASE REGISTER FOR 'EDISECT'
R15 ADDRESS OF 'EDSCAN'

EXIT CONDITIONS -

CC = 0 - NEW ARGUMENT FOUND WITH LENGTH 8
CC = 1 - NEW ARGUMENT FOUND WITH LENGTH 1-7
CC = 2 - NEW ARGUMENT FOUND WITH LENGTH 9 OR MORE
CC = 3 - END OF SCAN AREA REACHED

'EDISCRCH' BYTES 0-7 CONTAIN LEFT-JUSTIFIED PARM., BLANK-PADDED (OR TRUNCATED) TO 8 CHARACTERS...

R1 = ADDRESS OF NEW ARGUMENT (IF CC = 0, 1 OR 2)
R0 = LENGTH OF NEW ARGUMENT-1 (IF CC = 0, 1 OR 2)

LICENSED MATERIAL - PROPERTY OF IBM

PARAMETER LIST UPDATED TO REFLECT REMAING COUNT AND NEX
SCAN ADDRESS OF CC = 0, 1 OR 2; IF CC = 3, SCAN ADDRES
AND COUNT SET TO MAX. AND ZERO, RESPECTIVELY

ATTRIBUTES -

ENTERED VIA 'LINK' MACRO (USUALLY IN 'SCAN' MACRO')

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDCVN

FUNCTION -

PROVIDES GENERAL CONVERSION ROUTINES TO CONVERT

FROM BINARY TO HEX OR DECIMAL AND FROM HEX OR DECIMAL TO BINARY

ENTRY POINTS -

NUMBER CONVERSIONS:

EDCNVBH - TO CONVERT FROM BINARY TO PRINTABLE HEX
EDCNVBD - TO CONVERT FROM BINARY TO PRINTABLE DECIMAL
EDCNVHB - TO CONVERT FROM PRINTABLE HEX TO BINARY
EDCNVDB - TO CONVERT FROM PRINTABLE DECIMAL TO BINARY

CHARACTER STRING CONVERSIONS:

EDCNVIM - TO TEST FOR AND PERFORM IMPLICIT CONVERSION
OF A STRING FROM PRINTABLE HEX TO BINARY
OR FROM PRINTABLE BINARY TO BINARY

ENTRY CONDITIONS -

FOR NUMBER CONVERSIONS:

WHEN CONVERTING "TO" BINARY -
R0 AND R1 MUST CONTAIN PRINTABLE CHARACTERS
LEFT-JUSTIFIED AND PADDED WITH BLANKS
WHEN CONVERTING "FROM" BINARY -
R0 MUST CONTAIN BINARY VALUE TO BE CONVERTED
R11 MUST ALWAYS POINT TO 'EDISECT'
R15 MUST CONTAIN ENTRY POINT ADDRESS TO 'EDCNV'

FOR CHARACTER STRING CONVERSIONS:

R15 MUST CONTAIN ADDRESS OF 'EDCNVIM'
R11 MUST ALWAYS POINT TO 'EDISECT'
R10 MUST POINT TO CURRENT 'LVLSECT'
R1 CONTAINS ADDRESS OF STRING TO BE TESTED
R0 CONTAINS LENGTH OF STRING TO BE TESTED

EXIT CONDITIONS -

FOR NUMBER CONVERSIONS:

WHEN CONVERTING "TO" BINARY -

LICENSED MATERIAL - PROPERTY OF IBM

```
        R0 = BINARY VALUE (IF CC = 0)
        R1 = ZERO (IF CC = 0)
        R0 = UNDEFINED (IF CC ¬= 0)
        R1 = POINTER TO ILLEGAL CHARACTER (IF CC ¬= 0)
        CC = 0 - CONVERSION SUCCESSFUL
        CC ¬= 0 - ILLEGAL CHARACTER FOUND
    WHEN CONVERTING "FROM" BINARY -
        R0 AND R1 CONTAIN CONVERTED VALUE,
        RIGHT-JUSTIFIED AND LEFT-PADDED WITH ZEROS
        (IF CONVERTING TO HEX)
                    ** OR **
        R0 AND R1 CONTAIN CONVERTED VALUE,
        LEFT-JUSTIFIED AND RIGHT-PADDED WITH BLANKS
        (IF CONVERTING TO DECIMAL)

    FOR CHARACTER STRING CONVERSIONS:

        CC = 0 - IF STRING WAS CONVERTED
                 R1 POINTS TO RESULT STRING
                    (OVERLAYED ON INPUT STRING)
                 R0 CONTAINS LENGTH OF RESULT STRING
        CC = 1 - IF ERROR IN CONVERSION (BAD DATA)
                 R1 POINTS TO BAD CHARACTER
                    (PRIOR CHARACTERS ARE DESTROYED)
                 R0 CONTAINS REMAINING LENGTH
                    (NUMBER OF CHARACTERS NOT YET TESTED)
        CC = 3 - IF STRING DOES NOT IMPLY CONVERSION
                 (REGISTERS AND DATA REMAIN UNALTERED)


ATTRIBUTES -

    CALLED VIA 'LINK' MACRO
```

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDSTORE

FUNCTION -

TO MANAGE THE 'STORE' AND 'REC' BLOCKS

ENTRY POINTS -

EDSTOREG - TO 'GET' A 'REC' BLOCK
EDSTOREF - TO 'FREE' A 'REC' BLOCK
EDSTORER - TO 'RELEASE' ALL STORAGE ON A GIVEN LEVEL
EDSTOREX - TO 'GET' AN AUXILLARY BLOCK OF STORAGE

ENTRY CONDITIONS -

EDSTOREG -
R11 POINTS TO EDISECT
R10 POINTS TO LVLSECT
EDSTOREF -
R11 POINTS TO EDISECT
R10 POINTS TO LVLSECT
R1 POINTS TO 'REC' TO BE 'FREEED'
EDSTORER -
R11 POINTS TO EDISECT
R10 POINTS TO LVLSECT
EDSTOREX -
R11 POINTS TO EDISECT
R10 POINTS TO LVLSECT

EXIT CONDITIONS -

EDSTOREG -
R1 POINTS TO 'NEW' RECORD
R15 = 0 (AND CC = 0)
OR
R15 = 1 AND CC = 2 (IF NO STORAGE AVAILABLE
EDSTORER -
NONE
EDSTOREF -
R1 POINTS TO RECORD TO BE 'FREED'
EDSTOREX -
R1 POINTS TO 160-BYTE AUXILLARY STORAGE BLOCK
R15 (SAME AS 'EDSTOREG')

ATTRIBUTES -

ENTERED VIA 'CALL' MACRO

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDFNCADD

FUNCTION -

ADD 'N' RECORDS AFTER POINT 'P'

ENTRY CONDITIONS -

R11 = EDISECT
R10 = LVLSECT
R1 = PTR. TO LINE AFTER WHICH TO ADD
(OR 'LVLTOF' OR 'LVLEOF')
R0 = NUMBER OF LINES TO BE ADDED

EXIT CONDITIONS -

R0 - RESIDUAL COUNT OF ADDED LINES
R1 - PTR. TO LAST LINE ADDED
R15 - 1 AND CC = 2 - INSUFFICIENT STORAGE TO FULFILL REQUEST

OR

R15 - -1 AND CC = 1 - R0 ON ENTRY NOT GE 1 AND LE 65533 (R0 AND R1 UNCHANGED)

OPERATION -

1. IF AT TOF - LINES ADDED BEFORE FIRST LINE IN FILE
2. IF AT EOF - LINE PTR. SET TO LAST LINE IN FILE - THEN ADD FUNCTION PERFORMED
3. IF NO LINES IN FILE - PTR. SET TO TOF AND GO TO #1

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDFNCDEL

FUNCTION -

DELETE 'N' RECORDS STARTING AT POINT 'P'

ENTRY CONDITIONS -

R0 = NUMBER OF LINES TO BE DELETED
R1 = PTR. TO FIRST LINE TO BE DELETED
(OR 'LVLTOF' OR 'LVLEOF')

EXIT CONDITIONS -

R0 = RESIDUAL COUNT OF LINES DELETED
R1 = PTR. TO NEXT LINE AFTER LAST DELETED
R15 = 0 (AND CC = 0) - ALL REQUESTED LINES DELETED
R15 = 1 (AND CC = 2) - EOF REACHED BEFORE ALL REQUESTED LINES DELETED

OPERATION -

1. IF AT 'LVLTOF' - LINE PTR. SET TO FIRST LINE AND PROCEED TO # 3
2. IF AT 'LVLEOF' - NO DELETIONS MADE (R15 = 1 AND CC = 2)
3. DELETIONS PERFORMED BY DE-LINKING EACH RECORD AND RETURNING IT FOR 'SPARES' BY CALLING 'EDSTOREF'
4. OPERATION ENDS WHEN EOF IS REACHED. ON RETURN -
   R0 = RESIDUAL COUNT (# OF DELETIONS NOT PERFORMED)
   R1 = 'LVLEOF'
   R15 = 1 AND CC = 2

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDFNCFRD

FUNCTION -

READ SPECIFIED FILE INTO STORAGE

ENTRY CONDITIONS -

'EDIFCB' (EDIFN, EDIFT,EDIFM) MUST BE INITIALIZED WITH REQUESTED FILE ID

R1 - MUST POINT TO LINE AFTER WHICH NEW DATA IS TO BE READ.

R0 - B01 - FIRST ITEM NO. TO BE READ
B23 - NUMBER OF ITEMS TO BE READ

EXIT CONDITIONS -

IF CC = 0
R0 = 0
R1 = PTR. TO LAST ITEM READ
IF CC = 2
R0 = TOTAL COUNT OF ITEMS READ
R1 = PTR. TO LAST LINE READ
IF CC = 1
R0 AND R1 ARE UNCHANGED

CC = 0
R15 = 0 - ALL REQUESTED ITEMS HAVE BEEN READ
CC = 2
R15 = 1 - INSUFFICIENT STORAGE TO READ ENTIRE FILE
R15 = 2 - EOF REACHED BEFORE ALL ITEMS READ
CC = 1
R15 = -1 - ILLEGAL FILE ID (NAME, TYPE, AND/OR MODE)
R15 = -2 - DISK NOT CURRENTLY ACCESSED
R15 = -3 - FILE CANNOT BE READ (OPEN FOR OUTPUT OR PERMANENT I/O ERROR)
R15 = -4 - STARTING RECORD NUMBER EXCEEDS EOF
R15 = -5 - 'LRECL' OF REQUESTED FILE EXCEEDS 'LVLRECL' OR 'MAXSIZE'
R15 = -6 - FILE NOT FOUND

ERROR MESSAGE SET-UP IN EDIWORK BY 'EDFNCSTT'
IF RETURN CODE IS NEGATIVE

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDFNCWR

FUNCTION -

TO WRITE FILE RECORDS FROM STORAGE TO DISK

ATTRIBUTES -

RE-ENTRANT; ENTERED VIA 'CALL' MACRO

ENTRY CONDITIONS -

R11 - EDISECT BASE REG.
R10 - LVLSECT BASE REG.
R1 - ADDRESS OF FIRST RECORD TO BE WRITTEN
(OR 'LVLTOF' OR 'LVLEOF')
R0 - NUMBER OF RECORDS TO WRITE (IF PLUS)
OR
R0 - IF MINUS, SIGNALS REQUEST TO WRITE "UPDATES"
ONLY
'EDIFCB' (EDIFCBFN, EDIFCBFT, AND EDIFCBFM)
HAVE BEEN SET-UP WITH CORRECT FILE NAME, TYPE,
AND MODE, RESPECTIVELY

EXIT CONDITIONS -

CC = 0 - RECORDS SUCCESSFULLY WRITTEN
CC = 1 - NON-FATAL - END-OF-FILE BEFORE ALL
RECORDS WRITTEN
CC = 2 - NO RECORDS TO WRITE
(REQUEST TO START AT 'EOF')
(REQUEST TO WRITE ZERO RECORDS)
(FILE IS EMPTY)
('DBLANK' OPTION AND ALL RECORDS BLANK)
('UPDATE' WRITE REQUESTED BUT CHANGES NOT
BEING MAINTAINED)
CC = 3 - FATAL I/O ERROR FROM 'WRBUF'
('EDIBUFF' CONTAINS ERROR MESSAGE)
(R15 CONTAINS 'WRBUF' RETURN CODE)

REGISTER USAGE -

(IN WRITE ROUTINE - FROM LABEL 'WR1')

R2 = 'REC' POINTER
R3 = RESIDUAL RECORD COUNT
R4 = COUNT OF RECORDS IN BUFFER
R5 = BLOCKING FACTOR
R6 = DATA ADDRESS REGISTER
R7 = LENGTH OF THIS RECORD
R8 = 'LVLRECL' VALUE

LICENSED MATERIAL - PROPERTY OF IBM

R14 = CURRENT BUFFER ADDRESS
R15 = LENGTH OF 'TO' FOR 'MVCL'

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDFNCSTT

FUNCTION -

'STATE' THE FILE REFERENCED IN 'EDIFCB'

ENTRY CONDITIONS -

'EDIFCB' MUST BE FILLED-IN WITH THE
FILE NAME, TYPE, AND MODE TO BE VERIFIED

ATTRIBUTES -

RE-ENTRANT, ENTERED VIA 'CALL' MACRO

EXIT CONDITIONS -

R15 CONTAINS RETURN CODE FROM 'STATE'
R1 POINTS TO FST (IF CC = 0)

CONDITION CODE SET BY: LTR R15,R15

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDPNT

FUNCTION -

NAMED POINT DIRECTORY SERVICE ROUTINES.
TO PERFORM THOSE OPERATIONS NECESSARY TO
SCAN, ADD, AND DELETE ENTRIES FROM THE
DIRECTORY OF POINT NAMED RECORDS.

ENTRIES -

EDPNTLKP - GET 'REC' ADDR. CORRESPONDING TO
GIVEN "POINT" NAME.
EDPNTDEF - ASSIGN GIVEN POINT NAME TO SPECIFIED
'REC'.
EDPNTDEL - UNASSIGN THE SPECIFIED POINT NAME
(IF ASSIGNED).
EDPNTPRG - PURGE ALL REFERENCES TO THE SPECIFED
'REC'.
EDPNTLST - DISPLAY NAMES ASSIGNED TO A GIVEN
'REC'.

ENTRY CONDITIONS -

TO EDPNTLKP - 'WKD1' CONTAINS POINT NAME LEFT-
JUSTIFIED, AND PADDED WITH BLANKS
(AS RETURNED FROM 'EDSCANPR')

TO EDPNTDEF - 'WKD1' CONTAINS POINT NAME LEFT-
JUSTIFIED, AND PADDED WITH BLANKS
(AS RETURNED FROM 'EDSCANPR')
- R1 CONTAINS 'REC' ADDR. TO BE ASSIGNED

TO EDPNTDEL - 'WKD1' CONTAINS POINT NAME LEFT-
JUSTIFIED, AND PADDED WITH BLANKS
(AS RETURNED FROM 'EDSCANPR')

TO EDPNTLST - R1 CONTAINS 'REC' ADDR. FOR WHICH
TO SCAN.

TO EDPNTPRG - R1 CONTAINS 'REC' ADDR. TO BE PURGED

EXIT CONDITIONS -

FROM EDPNTLKP -
CC = 0 AND R15 = 0 - POINT FOUND - R1 CONTAINS
CORRESPONDING 'REC' ADDR.
CC ¬ 0 AND R15 = 1 - NOT FOUND - 'EDIBUFF'
CONTAINS ERROR MESSAGE

LICENSED MATERIAL - PROPERTY OF IBM

```
FROM EDPNTDEF -
      CC = 0 AND R15 = 0 - POINT DEFINED OR RE-
                           ASSIGNED.
      CC ¬= 0 AND R15 = 1 - NO ASSIGNMENT MADED -
                           'EDIBUFF' CONTAINS MESSAGE

FROM EDPNTDEL -
      CC = 0 AND R15 = 0 - NAMED POINT HAS BEEN
                           DELETED.
      CC ¬= 0 AND R15 = 1 - DELETION NOT PERFORMED.
                           'EDIBUFF' CONTAINS MESSAGE
FROM EDPNTLST -
      CC = 0 AND R15 = 0 - REFERENCES FOUND -
                           'EDIBUFF' CONTAINS DISPLAY
      CC ¬= 0 AND R15 = 1 - NO REFERENCES FOUND -
                           'EDIBUFF' CONTAINS MESSAGE

FROM EDPNTPRG -
      NO EXIT CONDITIONS REFLECTED
```

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDUPD

FUNCTION

THIS CONTROL SECTION HANDLES FILE UPDATING AND UPDATE GENERATION. THE TWO ENTRY POINTS, 'EDUPDI' AND 'EDUPDO', PERFORM UPDATING OF THE CURRENT FILE, AND CREATION OF AN UPDATE STREAM REFLECTING CURRENT FILE CHANGES, RESPECTIVELY. THE DETAILED OPERATION OF EACH IS DESCRIBED IN THE PROLOGUE PRIOR TO EACH ENTRY POINT.

ENTRY POINT 'EDUPD' CONVERTS THE CURRENT EDIT LEVEL TO A PROPER UPDATE "BASE". THAT IS, A "CHANGE" TABLE IS BUILT FOR THE CURRENT LEVEL (IF ONE ALREADY EXISTS IT IS PURGED) AND THE FILE RECORDS ARE VERIFIED FOR PRESENCE OF CORRECT SEQUENCE NUMBERS. THE CHANGE TABLE CONTAINS THE SEQUENCE FIELD (ALL 8 POSITIONS IN C.C. 73-80) FOR EACH ORIGINAL ('BASE') FILE RECORD. THE FIRST DOUBLE-WORD OF THE CHANGE TABLE CONTAINS THE NUMBER OF ENTRIES IN THE CHANGE TABLE (EQUAL TO THE NUMBER OF RECORDS IN THE 'BASE' FILE). AT THE SAME TIME, A POINTER TO THE CORRESPONDING CHANGE TABLE ENTRY IS PLACED IN THE 'BASE' FILE 'REC' AT THE END OF THE DATA PORTION OF THE RECORD. THE CHANGE TABLE POINTER (4-BYTES) WILL THEREFORE BE POSITIVE FOR ALL ENTRIES WHICH WERE IN THE ORIGINAL 'BASE' FILE AND HAVE NOT BEEN ALTERED. THOSE EDITING COMMANDS WHICH ALTER FILE DATA, MUST INDICATE THE CHANGE BY COMPLEMENTING THE CHANGE TABLE POINTER. RECORDS WHICH HAVE BEEN "MOVED" OR ADDED TO THE 'BASE' FILE WILL HAVE NO CHANGE TABLE POINTER (ZERO).

AFTER CONVERTING THE CURRENT EDIT LEVEL TO AN 'UPDATE' BASE, CONTROL IS TRANSFERRED TO 'EDUPDI' TO PERFORM THE UPDATING OF THE CURRENT FILE. THEREFORE, ENTRY CONDITIONS AND EXIT CONDITIONS FOR ENTRY POINT 'EDUPD' ARE EXACTLY AS THOSE FOR 'EDUPDI' (SEE PROLOGUE BELOW) EXCEPT THAT ADDITIONAL ERROR MESSAGES MAY BE PRODUCED FOR ERRORS ENCOUNTERED IN ESTABLISHING THE EDIT LEVEL 'BASE' FILE.

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDUPDI

FUNCTION -

UPDATE A GIVEN FILE LEVEL FROM A
STANDARD CMS UPDATE CONTROL FILE

ENTRY CONDITIONS -

EDIFCB ('EDIFCBFN', 'EDIFCBFT', AND
'EDIFCBFM') MUST CONTAIN FILE NAME,
TYPE, AND MODE, RESPECTIVELY, OF THE
UPDATE FILE.

R11 - ADDRESS OF 'EDISECT'
R10 - ADDRESS OF 'LVLSECT'

EXIT CONDITIONS -

CC = 0 - UPDATE SUCCESSFULLY PERFORMED
CC = 1 - UPDATE NOT PERFORMED -
UPDATE FILE NOT FOUND
CC = 2 - UPDATE NOT PERFORMED -
INSUFFICIENT STORAGE FOR UPDATE
CC = 3 - UPDATE UNSUCCESSFUL -
'EDIBUFF' CONTAINS ERROR MSG.

METHOD OF OPERATION -

1. THE UPDATE FILE IS READ (VIA CALL TO 'EDFNCFRD' AND APPENDED TO THE END OF THE CURRENT FILE.
2. THE UPDATE STREAM IS SCANNED AND A CHAIN OF UPDATE CONTROL CARDS IS BUILT ANCHORED TO 'SAVWRK1'. ALL CONTROL CARDS AND INPUT ARE VERIFIED FOR FORMAT AND SEQUENCE AND P-LIST INFORMATION IS PLACED IN EACH CONTROL CARD IN THE FORMAT LISTED UNDER "DATA AREAS USED" BELOW. IF AUTOMATIC RESEQUENCING OF INPUT WAS REQUESTED, THE CARDS ARE RESEQUENCED IN THIS STEP AS THEY ARE SCANNED.
3. IF STEP 2 WAS SUCCESSFULLY COMPLETED THE CHAIN OF CONTROL CARDS IS EXECUTED ONE AT A TIME UNTIL UPDATE IS COMPLETE. FOR AN INSERT CARD, THE RECORDS TO BE INSERTED ARE UNCHAINED FROM THE UPDATE STREAM AND CHAINED INTO THE FILE DATA

LICENSED MATERIAL - PROPERTY OF IBM

IN THE SAME FASHION AS WITH THE "MOVE" COMMAND. FOR A DELETE CARD, 'EDFNCDEL' IS CALLED TO DELETE THE APPROPRIATE RECORDS. A REPLACE CARD, IS HANDLED AS AN INSERT FOLLOWED BE A DELETE.

DATA AREAS USED -

THE FOLLOWING INFORMATION WILL BE STORED IN EACH CONTROL CARD DEPENDING ON TYPE

CC	FIELD	DESCRIPTION
01-04	RECDATA+0	START SEQ. # FOR RESEQ.
05-08	RECDATA+4	SEQ. INCR. # FOR RESEQ.
09-12	RECDATA+8	DEL/REP - PRIOR 'REC'
53-56	RECDATA+52	'SEQNO1' "REC" ADDRESS
57-60	RECDATA+56	# OF RECS TO DELETE
61-64	RECDATA+60	FIRST ADDED 'REC' ADDR
65-68	RECDATA+64	LAST ADDED 'REC' ADDR.
69-72	RECDATA+68	PTR. TO NEXT CTL. CARD
73-74	RECDATA+72	CONTROL CARD TYPE CODE './ I ' = 4 './ D ' = 8 './ R ' = 12
75-75	RECDATA+74	INPUT RESEQUENCING FLAG X'00' - NO RESEQUENCING X'FF' - RESEQUENCE INP.

'SAVWRK1' CONTAINS CONTROL CARD CHAIN ANCHOR
'SAVWRK2' CONTAINS CHAIN "LAST" CARD POINTER
'SAVWRK3' POINTS TO FIRST "REAL" CTL. CARD
'WKD1' HOLDS THE SCAN ARGUMENT USED WHILE SCANNING UPDATE CONTROL CARD
'WKD5' BYTE 0 IS SEQ. FLAG. IF ZERO, NO './ S ' PROVIDED; IF X'FF', './ S' HAS BEEN READ; IF X'80', './ S' IS NOT PERMITTED (NOT FIRST CTL. CARD). BYTES 1-3 CONTAIN 3-CHAR. SEQ. LABEL (IF 'SEQ8' NOT SPECIFIED).
'WKD6' BYTES 0-3 CONTAIN SEQ. START NUMBER
BYTES 4-7 CONTAIN SEQ. INCREMENT VALUE
'WKD7' CONTAINS THE CURRENT SCAN POINTERS (USED DURING CONTROL CARD SCAN)
'WKD8' CONTAINS THE COMPARE INSTRUCTION USED TO SEARCH FOR REFERENCED SEQUENCE NUMBERS

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDUPDO

FUNCTION -

GENERATE AN UPDATE DATA STREAM TO REFLECT CHANGES IN THE CURRENT FILE RELATIVE TO THE ORIGINAL MASTER (SOURCE) FILE.

ENTRY CONDITIONS -

EDIFCB ('EDIFCBFN', 'EDIFCBFT', AND 'EDIFCBFM') MUST CONTAIN FILE NAME, TYPE, AND MODE, RESPECTIVELY, OF THE UPDATE FILE.

R11 - ADDRESS OF 'EDISECT'
R10 - ADDRESS OF 'LVLSECT'

EXIT CONDITIONS -

CC = 0 - UPDATE FILE WRITTEN SUCCESSFULLY
CC = 1 - UPDATE NOT WRITTEN -
CHANGES NOT BEING MAINTAINED
CC = 2 - 'WRBUF' ERROR OCCURRED -
(R15 CONTAINS ERR. RETURN CODE)
CC = 3 - FILE CANNOT BE WRITTEN -
'EDIBUFF' CONTAINS ERROR MSG.

METHOD OF OPERATION -

1. AN ATTEMPT IS MADE TO ALLOCATE AN OUTPUT BUFFER WHICH CAN HANDLE TWO OR MORE CARD IMAGES. IF THE BUFFER IS OBTAINED, THE MAXIMUM BLOCKING FACTOR IS CALCULATED AND 'EDIFCB' IS COMPLETED WITH THE NECESSARY INFORMATION TO WRITE THE FILE. IF STORAGE IS NOT AVAILABLE 'LVLDUMMY' IS USED TO BUILD THE OUTPUT RECORDS AND AS THE OUTPUT BUFFER FOR UN-BLOCKED OUTPUT.
2. THE UPDSCT (UPDATE SEGMENT CONTROL TABLE) IS INITIALIZED AND THE FILE IS SCANNED (STARTING AT TOF) UNTIL ONE OF THE FOLLOWING EVENTS OCCURS:
   A. AN "ADDED" RECORD IS FOUND
   B. SOURCE RECORDS ARE FOUND TO HAVE BEEN DELETED
   C. A RECORD IS FOUND TO HAVE BEEN REPLACED

LICENSED MATERIAL - PROPERTY OF IBM

D. 'EOF' IS REACHED OR END OF CHANGE TABLE IS REACHED.

3. WHEN ANY OF THE ABOVE OCCURS, CONTROL IS TRANSFERRED TO A ROUTINE WHICH UPDATES THE UPDSCT FOR THIS SEGMENT.
4. WHEN A SEGMENT IS COMPLETE (NO CHANGED RECORDS OR END-OF-FILE ENCOUNTERED), THE UPDSCT IS USED TO BUILD THE UPDATE CONTROL CARD FOR THIS SEGMENT AND THE UPDSCT SUB-FIELDS USED TO SEQUENCE AND OUTPUT THE NECESSARY DATA RECORDS.

DATA AREAS USED -

THE 'UPDSCT' (UPDATE SEGMENT CONTROL TABLE IS BUILT IN 'EDIWORK' AND HAS THE FOLLOWING FORMAT.

BYTES 0-3 CONTAIN CHANGE TABLE POINTER FOR THE FIRST DELETED ITEM IN THIS SEGMENT (OR ZERO IF NONE DELETED)

BYTES 4-7 CONTAIN CHANGE TABLE POINTER FOR THE LAST DELETED ITEM IN THIS SEGMENT (OR ZERO IF NONE DELETED)

BYTES 8-(END-OF-'EDIWORK') - CONTAIN A SERIES OF VARIABLE LENGTH SUB-FIELDS. THE TYPES OF SCT SUB-FIELDS CURRENTLY DEFINED AND THEIR FORMATS ARE:

A. 'ADD' SUB-FIELD:
BYTE 0 = 4 (INDICATES ADD TYPE)
1-3 = FIRST ADDED 'REC' ADDR.
4-7 = NUMBER OF RECS ADDED
8-11 = CTP FOR START SEQ. #
12-15 = SEQUENCING INTERVAL

B. 'CHANGED' SUB-FIELD:
BYTE 0 = 8 (INDICATES CHG TYPE)
1-3 = FIRST CHANGED REC ADDR
4-7 = NUMBER OF CHANGED RECS.

'SAVWRK1', 'SAVWRK2', AND 'SAVWRK3' ARE USED TO HOLD THE BLOCKING POINTERS FOR THE UPDATE FILE WRITING ROUTINES:

'SAVWRK1' CONTAINS ADDR. OF NEXT BUFFER LOCATION
'SAVWRK2' CONTAINS ADDR. OF LAST BUFFER LOCATION
'SAVWRK3' ADDRESS OF THE BUFFER
'WKD6' CONTAINS THE ADDRESS AND NUMBER OF DOUBLE-WORDS IN THE BUFFER OBTAINED OR ZERO IF NO BUFFER WAS OBTAINED.
'WKD7' AND 'WKD8' ARE USED AS A SAVE AREA FOR

LICENSED MATERIAL - PROPERTY OF IBM

REGISTERS R14-R1 AFTER SEGMENT
CONTROL TABLE COMPLETION AND
DURING DATA OUTPUT.

LICENSED MATERIAL - PROPERTY OF IBM

EDIT3

FUNCTION -

THIS SOURCE MODULE COMPRISES THE MAJOR FUNCTIONS OF THE EDIT COMPONENT OF THE SYSTEM.

CONTENTS

THE FOLLOWING CONTROL SECTIONS ARE INCLUDED IN THIS MODULE:

EDLVL - ENTERS A NEW OR EXISTING EDIT LEVEL
EDPROF - READS, ANALYZES AND STORES THE CONTENTS OF THE $PROFILE DATA SET.
EDLDB - FORMATS A SET OF LOGICAL DISPLAY BLOCKS (LDB'S) AND SET THE CORRESPONDING SCREEN FORMAT.
EDGAR - PERFORMS EDITING SYSTEM INITIALIZATION AND TERMINATION

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDLVL

FUNCTION -

TO PARSE THE LEVEL CREATION COMMAND, OBTAINING DEFAULTS FROM THE PROFILE, AND TO BUILD AND INTIALIZE A NEW 'LVLSECT' CONTROL BLOCK FOR THE SPECIFIED FILE OR ACCESS A PREVIOUSLY BUILT 'LVLSECT' CONTROL BLOCK TO CONTINUE EDITING.

ENTRY CONDITIONS -

R10	-	POINTS TO "CURRENT" 'LVLSECT' (OR ZERO IF NONE)
R0-R1	-	CONTAIN CURRENT SCAN POINTERS TO REMAINDER OF COMMAND LINE REQUESTING NEW LEVEL.
'WKD9'	-	NAME OF COMMAND WHICH WAS ISSUED.

EXIT CONDITIONS -

R10 - CONTAINS NEW (OR SAME) "CURRENT" EDIT LEVEL 'LVLSECT' ADDRESS

R0 - R1 - UPDATED TO REFLECT END OF SCAN OR POSITION AT WHICH SCAN STOPPED IN THE EVENT OF ERRONEOUS OPERAND.

CC = 0 - NEW LEVEL CUREATED - R10 CONTAINS ADDRESS OF CURRENT 'LVLSECT'
CC = 1 - NON-FATAL ERROR - R10 CONTAINS ADDRESS OF CURRENT 'LVLSECT' AND 'EDIBUFF' CONTAINS APPROPRIATE ERROR MESSAGE
CC = 3 - FATAL ERROR - R10 REMAINS UNCHANGED AND 'EDIBUFF' CONTAINS ERROR MESSAGE.

TABLES/WORK AREAS USED -

'WKD4-WKD8' - WILL HOUSE THE 'ENTER' COMMAND PARM. LIST

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDPROC

FUNCTION -

TO READ, ANALYZE, AND STORE THE INFORMATION PROVIDED IN AN EDGAR '$PROFILE' FILE.

ENTRY CONDITIONS -

R11 - 'EDISECT' BASE
'EDIFCBFN' - CONTAINS THE FILENAME OF THE FILE (WHOSE FILETYPE IS $PROFILE) WHICH CONTAINS THE PROFILE DEFINITION STATEMENTS.

EXIT CONDITIONS -

CC = 0 - PROFILE READ AND STORED
CC = 1 - PROFILE FILE DOES NOT EXIST
CC = 3 - ERROR IN PROFILE READING OR ANALYSIS

'EDIBUFF' CONTAINS APPROPRIATE ERROR MESSAGE FOR CONDITION CODES 1 AND 2.

TABLES/WORKAREAS USED -

'WKD1' - 'WKD9' - USED DURING PARSING PHASES FOR PARSING INFORMATION AND TEMPORARY PROFILE.

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDLBD

FUNCTION -

TO SET-UP LOGICAL DISPLAY BLOCKS FOR AN A GIVEN EDIT LEVEL OR TO RESET THE CURRENT DISPLAY BLOCKS BASED ON FORMATTING INFORMATION SUPPLIED AT ENTRY. THEN, TO FORMAT THE SCREEN IMAGE ACCORDING TO INFORMATION IN THE LOGICAL DISPLAY BLOCKS.

ENTRY -

R11 - STANDARD ADDR. OF EDISECT
R10 - STANDARD ADDR. OF LVLSECT
'LVLFMTHW' - FIRST HALFWORD - CONTAINS THE NET DIFFERENCE IN NUMBER OF LOGICAL DISPLAYS. (ZERO INDICATES NO NET DIFFERENCE; A NEGATIVE VALUE INDICATES '-N' DISPLAYS TO BE RELEASED; A POSITIVE VALUE INDICATES '+N' DISPLAYS TO BE ADDED TO THE CHAIN.)

'LVLFMTHW' - REMAINING HALFWORDS - CONTAINS THE FORMAT HALF-WORDS (ONE FOR EACH LOGICAL DISPLAY WHICH WILL EXIST)

EACH FORMAT HALF-WORD CONTAINS THE FOLLOWING INFO:
BYTE 0 - NUMBER OF SCREEN LINES PER DATA-DISPLAY BUFFER) - SEE NOTE 1 BELOW
BYTE 1 - NUMBER OF DATA-DISPLAY BEFFERS IN THIS DISPLAY

EXIT -

CC = 0 - SCREEN AND LDB'S HAVE BEEN FORMATTED
CC = 3 - INSUFFICIENT STORAGE FOR INITIALIZATION

NOTES -

1. IF THE HIGH-ORDER BIT OF BYTE 0 IN ANY FORMAT HALFWORD IS 'ON', THE CORRESPONDING LOGICAL DISPLAY IS TO CONTAIN TYPE III AREAS.

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDGAR

FUNCTION -

TO INITIALIZE THE CONTROL BLOCKS AND THE FIRST EDIT LEVEL AND TO TRANSFER TO THE BASIC SCREEN OPERATION MANAGER FOR RUNNING CONTROL.

TO PROVIDE GENERAL TERMINATION HANDLING INCLUDING THE RELEASE OF ALL FREE STORAGE CONTROL BLOCKS AND TO RETURN TO THE CALLING SYSTEM COMPONENT (USUALLY DMSINI).

ENTRY CONDITIONS -

R15 - CONTAINS THE ENTRY POINT ADDRESS (EDGAR)
R14 - CONTAINS THE RETURN ADDRESS
R13 - POINTS TO AN 8 DOUBLE-WORD SAVE AREA
R1 - POINTS TO A PARAMETER LIST IDENTICAL TO THAT USED FOR 'ENTER' COMMAND

EXIT CONDITIONS -

R0 - R14 - REMAIN UNCHANGED
R15 - CONTAINS A RETURN CODE (ZERO IF NO ERRORS)

OPERATION -

1. SAVE ENTRY REGISTERS AND CALL 'DMSFREE' TO OBTAIN THE 'EDISECT' CONTROL BLOCK. CLEAR THE BLOCK TO ZEROS. ALLOW FREE STORAGE ABEND IF INSUFFICIENT STORAGE TO ALLOCATE 'EDISECT'.
2. INITIALIZE THE INTERNAL SAVE AREA CHAIN (WITHIN 'EDISECT') TO ALLOW INTERNAL 'CALL'S TO BE ISSUED.
3. INITIALIZE THE 'IOSECT' CONTROL BLOCK BY CALLING "EDIOINIT".
4. CHECK FOR THE EXISTENCE OF FILE 'EDIT CMSUT1' ON A R/W DISK. IF IT EXISTS, TERMINATE IMMEDIATELY.
5. LOCATE END OF INPUT PARAMETER LIST, POINT TO START AND END AND CALL 'EDLVL' TO GENERATE THE INITIAL EDIT LEVEL.
6. 'EDTERM' IS THE GENERAL TERMINATION ENTRY POINT. WHEN CONTROL IS TRANSFERRED TO THIS POINT (BY 'EDSCR'), RETURN ALL CONTROL BLOCKS TO FREE STORAGE, RESTORE THE INITIAL ENTRY SAVE AREA ADDRESS, RESTORE THE INITIAL ENTRY REGISTERS, AND RETURN TO THE CALLING PROGRAM WITH THE RETURN CODE (FROM 'EDIRETC') IN REGISTER 15.

LICENSED MATERIAL - PROPERTY OF IBM

TABLES/WORKAREAS USED -

'EDISECT' AND 'IOSECT' ARE OBTAINED DURING INITIALIZATION AND RETURNED DURING TERMINATION.

'EDIFILES' CHAIN IS BUILT DURING INITIALIZATION AND RETURNED AT TERMINATION. THIS CHAIN CONSISTS OF A SERIES OF SINGLE-CHAINED ENTRIES EACH 3 DOUBLE-WORDS LONG. THE FIRST FULL-WORD CONTAINS THE ADDRESS OF THE NEXT CHAIN ELEMENT. THE SECOND FULL-WORD IS UNUSED. THE SECOND DOUBLE-WORD CONTAINS THE NAME OF THE DEFAULT FILETYPE AND THE THIRD DOUBLE-WORD, THE NAME OF THE AUTO-EXEC TO BE INVOKED WHEN INITIALIZING A LEVEL OF THIS TYPE.

LICENSED MATERIAL - PROPERTY OF IBM

FUNCTION -

TO MAINTAIN CONTINUITY OF OPERATION OF THE PHYSICAL VIRTUAL SCREEN AND LEVEL VIRTUAL SCREEN; ALSO, CONTAINS THE SCREEN OPERATION SIMULATION FACILITY (SOS) AND PERFORMS PHYSICAL 3270 INPUT/OUTPUT PROCESSING.

CALLED BY -

THIS MODULE ACTS AS THE PRIMARY DRIVER. THEREFORE, CONTROL IS TRANSFERRED TO IT FROM 'EDGAR' (EDITOR INITIALIZATION SECTION) IMMEDIATELY AFTER THE FIRST EDIT LEVEL HAS BEEN CREATED (INITIALIZATION IS COMPLETE).

ENTRY CONDITIONS -

R11 - 'EDISECT' BASE REGISTER

REGISTER USAGE -

R13 - STD. REG. SAVE AREA BASE
R12 - MODULE BASE REGISTER
R11 - 'EDISECT' BASE REG.
R10 - 'LVLSECT' BASE REG.
R9 -'IOSECT' BASE REG.

DESCRIPTION -

THIS MODULE CONTAINS THE FOLLOWING SIX INDPENDENT CONTROL SECTIONS (CSECTS):

1. EDIO - CONTAINS THE EDGAR SCREEN MANAGER AND FRONT END DRIVER LOGIC.
2. EDIOA - CONTAINS THE SCREEN OPERAION SIMULATION FACILITY (SOS).
3. EDIOB - CONTAINS THE PHYSICAL 3270 INPUT/OUTPUT ROUTINES.
4. EDIOC - CONTAINS TABLES USED BY OTHER CONTROL SECTIONS.
5. EDIOINIT - PERFORMS INITIALIZATION OF THE I/O AND FRONT-END CONTROL TABLES.
6. EDIOSVC - CMS-TO-EDGAR SVC (X'ED') INTERFACE ROUTINES.

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDIO

ENTRY CONDITIONS -

THIS CONTROL SECTION IS ENTERED AT SYSTEM INITIALIZATION IMMEDIATELY FOLLOWING THE FIRST EDIT LEVEL INITIALIZATION. IT REMAINS IN CONTROL THROUGHOUT THE EDGAR SESSION AND DOES NOT EXIT.

OPERATION -

THE CSECT IS COMPRISED OF THREE BASIC ROUTINES:

1. LEVEL-TO-PHYSICAL VIRTUAL SCREEN MAPPING;
2. PHYSICAL-TO-LEVEL VIRTUAL SCREEN UNMAPPING;
3. PROCESS AND DISPLAY DRIVERS.

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDIOA

FUNCTION -

THIS CONTROL SECTION CONTAINS THE SCREEN OPERATION SIMULATION (SOS) FACILITIES. IT PERFORMS THE FUNCTIONS INDICATED IN BY A PRE-PROGRAMMED SET OF INSTRUCTIONS READ FROM THE CMS STACK.

IN ADDITION, THE 'KEYBOARD FUNCTION' ROUTINES DEFINED FOR EDGAR ARE CONTAINED HEREIN. THESE WILL BE ENTERED BOTH AS A RESULT OF 'SOS' COMMANDS OR DEPRESSION OF THE CORRESPONDING KEY ON THE PHYSICAL 3277 DISPLAY STATION.

ENTRY CONDITIONS -

THIS CONTROL SECTION IS LOGICALLY PART OF THE STANDARD FRONT END EDGAR I/O PROCESSOR. IT RECEIVES CONTROL FROM 'EDIO' (THE SCREEN MANAGER AND DRIVER CSECT) WHEN THE PHYSICAL VIRTUAL SCREEN ('IOSCREEN') HAS BEEN BUILT AND NO ERRORS EXIST THEREON OR '/SOS ERROR CONTINUE' HAS BEEN SET. IN ADDITION, ANY OF ITS 'ACTION KEY' ROUTINES (E.G. 'ENTER' 'PF1'-'PFNN', 'PA1', 'PA2', ETC.) MAY BE ENTERED DIRECTLY WHEN THE FUNCTION INDICATED BY EACH IS TO BE PERFORMED.

OPERATION -

PROCESSING BEGINS AT LABEL 'IO' TO READ FROM THE CMS STACK. PROCESSING CONTINUES UNTIL THE STACK IS EMPTY OR UNTIL A NULL LINE IS ENCOUNTERED IN THE STACK. AT THIS TIME, CONTROL IS PASSED TO THE PHYSICAL 3277 INPUT/OUTPUT CONTROL SECTION, 'EDIOB'.

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDIOB

FUNCTION -

THIS MODULE WILL GENERATE A DATA STREAM WHICH CAUSES THE CURRENT CONTENTS OF THE PHYSICAL VIRTUAL SCREEN ('IOSCREEN') TO BE REPLICATED AT THE REAL 3277 DISPLAY STATION. TRANSMISSION OPTIMIZATION IS ATTEMPTED BY SENDING ONLY THOSE FIELDS WHICH HAVE BEEN CHANGED ('FLDCHG' BIT SET) AND, FOR REMOTELY CONNECTED DEVICES, COMPRESSING FIVE OR MORE DUPLICATE CHARACTERS TO A SINGLE REPEAT-TO-ADDRESS ORDER.

ENTRY CONDITIONS -

THIS CSECT IS LOGICALLY PART OF THE EDGAR FRONT-END MANAGEMENT ROUTINES (MODULE 'EDIO') AND IS CALLED WHEN NORMAL FRONT-END PROTOCOL REQUIRES THE PHYSICAL 3277 TO BE UPDATED.

ENTRY EDIOC

FUNCTION -

TO INITIALIZE THE PRIMARY CONTROL BLOCKS USED BY THE I/O PROCESSOR COMPONENT (NAMELY, 'IOSECT').

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDSVC

FUNCTION -

THIS ROUTINE HANDLES THE X'ED' SVC WHILE EDGAR IS RUNNING. THIS SVC PERMITS A COMMAND (PROBABLY RUNNING IN THE SUBSET) TO ISSUE AN EDGAR COMMAND INTERNALLY AND RECEIVE THE RESULTS OF THE COMMAND (RETURN CODE AND POSSIBLE OUTPUT).

ENTRY CONDITIONS -

R14 - CONTAINS RETURN ADDRESS TO CMS' SVC HANDLING ROUTINES
R13 - CONTAINS ADDRESS OF AN 18 FULL-WORD SAVE AREA (SET-UP BY CMS)
R12 - ADDRESS OF THIS ROUTINE (SET-UP BY CMS)
R1 - POINTS TO A 132-BYTE COMMAND AREA (SET BY ISSUER OF THE SVC). THE AREA MUST CONTAIN THE COMMAND TO BE HANDLED AND MUST BE PADDED TO ITS FULL LENGTH WITH BLANKS. CHARACTERS ARE NOT TRANSLATED OR CONVERTED IN ANY FASHION (THIS MUST BE DONE, IF NECESSARY, BY CALLING ROUTINE).

EXIT CONDITIONS -

R15 - CONTAINS RETURN CODE AS FOLLOWS:
0 - COMMAND SUCCESSFULLY EXECUTED
>0 - ERROR - COMMAND CANNOT BE PROCESSED; R15 CONTAINS THE RETURN CODE (SAME AS THE ERROR MESSAGE NUMBER). THE CALLER'S COMMAND AREA CONTAINS THE ERROR MESSAGE.
-4 - NO ERROR - BUT COMMAND HAS GENERATED OUTPUT. THE CALLER'S COMMAND AREA CONTAINS THE OUTPUT PRODUCED BY THE COMMAND.

NOTES:

1. THE INTERFACE IS ACTIVATED ONLY DURING EDIT PROCESSING. THUS, THE MODULE ISSUING THE EDIT COMMAND SVC MUST BE PASSED CONTROL BY AN EDIT PROCESSING ROUTINE (SUCH AS THE COMMAND HANDLER). THIS REQUIREMENT EXISTS BECAUSE OF THE NEED TO MAINTAIN THE STATUS

LICENSED MATERIAL - PROPERTY OF IBM

OF THE SCREEN ON ANY GIVEN EDIT LEVEL DURING I/O PROCESSING (RELATIVE TO CURRENT LINE, LINES DISPLAYED, ETC.).

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDCMDO

FUNCTION -

THIS CONTROL SECTION IS ENTERED TO PROCESS TYPE I AND TYPE II COMMANDS FOR ANY LOGICAL DISPLAY. ON ENTRY, THE LOGICAL DISPLAY COMMAND BUFFER ('LDBCMD') CONTAINS THE COMMAND TO BE PROCESSED. THIS MODULE WILL ANALYZE THE COMMAND, LOCATE THE COMMAND HANDLER AND TRANSFER CONTROL TO IT. THIS CSECT REMAINS ADDRESSABLE DURING ALL TYPE I/TYPE II COMMAND PROCESSING (SEPARATE ADDRESSABILITY IS PROVIDED TO THE CSECT CONTAINING THE SPECIFIC COMMAND ANALYZER). RETURNS ARE MADE TO LOCATIONS WITHIN THIS CONTROL SECTION TO TERMINATE PROCESSING, RETURN RESULTANT DATA, OR RETURN AN ERROR MESSAGE. ALL ERROR MESSAGE GENERATION AND ACCESS IS PERFORMED WITHIN THIS CONTROL SECTION.

ENTRY CONDITIONS

R15 = ENTRY POINT ADDRESS ('EDCMD')
R14 = RETURN ADDRESS
R13 = STANDARD EDITOR SAVE AREA
R11 = 'EDTSECT' BASE
R10 = 'LVLSECT' BASE
R9 = 'LDBSECT' BASE FOR LOGICAL DISPLAY TO BE PROCESSED.

'LDBCMD' CONTAINS THE COMMAND TO BE PROCESSED. THIS AREA MUST BE PADDED-RIGHT WITH BLANKS, IF NECESSARY, AND ANY TRANSLATION MUST PREVIOUSLY HAVE BEEN PERFORMED. NOTE: INDIVIDUAL TOKENS WILL BE FORCED TO UPPER CASE FOR THE PURPOSE OF EXAMINATION ONLY.

EXIT CONDITIONS

R15 AND 'LDBRETCD' CONTAIN THE COMMAND RETURN CODE AS FOLLOWS:

RC = 0 - COMMAND SUCCESSFULLY PROCESSED ('LDBCMD' CONTAINS BLANKS).
RC = -1 - COMMAND SUCCESSFULLY

LICENSED MATERIAL - PROPERTY OF IBM

PROCESSED ('LDBCMD' CONTAINS RESULT)

RC > 0 - ERROR ENCOUNTERED ('LDBCMD' CONTAINS AN APPROPRIATE ERROR MSG.) RETURN CODE IS THE ERROR MESSAGE NUMBER (500 OR GREATER).

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDCMD1

FUNCTION -

THIS CONTROL SECTION COMPRISES THE FIRST SECTION OF GENERAL COMMAND PROCESSORS.

CONTENTS -

THE PRIMARY PROCESSING ROUTINES FOR THE FOLLOWING COMMANDS ARE INCLUDED IN THIS CONTROL SECTION:

1. TOP
2. BOTTOM
3. UP
4. DOWN
5. FORWARD
6. BACKWARD
7. DELETE
8. REPLACE
9. INPUT
10. INSERT
11. ADD
12. DUP
13. STACK
14. CASE
15. POINT
16. LEFT
17. RIGHT
18. TABS
19. VIEW
20. MOVE
21. COPY

ENTRY EDCMD2

FUNCTION -

THIS CONTROL SECTION COMPRISES THE SECOND SECTION OF TYPE I/TYPE II COMMAND PROCESSORS.

CONTENTS -

THE PRIMARY PROCESSING ROUTINES FOR THE FOLLOWING TYPE II COMMANDS ARE INCLUDED IN THIS CONTROL SECTION:

1. FIND
2. LOCATE
3. SEARCH
4. ZONE
5. CHANGE
6. SHIFT
7. DSPC
8. RENUM

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDCMD3

FUNCTION -

THIS CONTROL SECTION COMPRISES THE THIRD SECTION OF TYPE I/TYPE II COMMAND PROCESSORS.

CONTENTS -

THE PRIMARY PROCESSING ROUTINES FOR THE FOLLOWING TYPE I/TYPE II COMMANDS ARE INCLUDED IN THIS CONTROL SECTION:

1. DSPF
2. PROFILE
3. RECFM
4. SET
5. SERIAL
6. ENTER
7. EXEC
8. SCREEN
9. FORMAT
10. FILE
11. SAVE
12. END
13. QUIT
14. CANCEL
15. PF

LICENSED MATERIAL - PROPERTY OF IBM

ENTRY EDIMSG

FUNCTION -

THIS MODULE CONTAINS NO EXECUTABLE CODE.
THE TEXT OF ALL ERROR MESSAGES PRODUCED BY
EDGAR IS CONTAINED HEREIN. CERTAIN MESSAGES
REQUIRE DATA TO BE SUBSTITUTED WHICH IS
ACQUIRED DYNAMICALLY DURING THE EDITING PROCESS.

LICENSED MATERIAL - PROPERTY OF IBM

EDISECT

FORMAT OF MAIN EDITOR CONTROL BLOCK

```
EDISECT  DSECT  ,              START PRIMARY EDITOR DATA SECTION
EDIBUFF  DS     CL2304         GENERAL EDITOR I/O BUFFER
EDIWORK  EQU    EDIBUFF        ALSO USED FOR GENERAL WORK PURPOSES
EDISCRCH DS     9D             'SCRATCH' AREA FOR GENERAL WORK PURPOSES
EDITLVL  DS     1F             CURRENTLY ACTIVE 'LVLSECT' PTR.
EDIRETA  DS     1F             EDITOR RETURN ADDRESS TO CMS
EDIRETC  DS     1H             CMS' RETURN CODE FROM EDITOR
EDIDSPQ# DS     1H             NUMBER OF ENTRIES IN DISPLAY QUEUE
EDIPROF  DS     1F             ANCHOR TO CHAIN OF FILETYPE PROFILES
EDIOSECT DS     1F             ADDRESS OF I/O CONTROL SECTION
EDICULDB DS     1F             CURRENT 'LDBSECT' CELL (FOR SVC)
EDITSCR  DS     12F            LOGICAL SCREEN BUCKETS
EDITSCRN EQU    *              END OF LOGICAL SCREEN TABLE
EDITDSPQ DS     24F            QUEUE OF LEVELS ELIGIBLE FOR DISPLAY
EDIDSPQX EQU    (*-EDITDSPQ)/L'EDITDSPQ MAX. ENTRIES IN DISPLAY QUEUE
EDIAUTID DS     1F             LATEST USED AUTOSAVE ID
         DS     1H             RESERVED
EDIFLG1  DS     1X             EDITOR FLAGS - BYTE 1
EDIFLG2  DS     1X             EDITOR FLAGS - BYTE 2
EDICELL1 DS     CL120          TEMPORARY COMMAND WORKING CELL
EDIFCB   DS     0D             START OF GENERAL-PURPOSE EDITOR I/O FCB
EDIFCBCM DS     CL8            CMS FUNCTION
EDIFCBID DS     0CL18          FILE ID FOR I/O
EDIFCBFN DS     CL8            FILE NAME
EDIFCBFT DS     CL8            FILE TYPE
EDIFCBFM DS     CL2            FILE MODE
EDIFCBFR DS     1H             FIRST RECORD NUMBER
EDIFCBBA DS     1A             ADDRESS OF I/O AREA....
EDIFCBBS DS     1F             SIZE OF I/O AREA
EDIFCBFV DS     CL2            FIXED/VARIABLE FLAG...
EDIFCBBF DS     1H             BLOCKING FACTOR
EDIFCBCT DS     1F             COUNT OF BYTES READ
EDISAVE  DS     1F             PTR. TO CHAIN OF FREE SAVE AREAS
EDISAVEA DS     (TTLSAVE)XL(SAVEAREL*8) INTERNAL SAVE AREAS
         CNOP   0,8            ALIGN END TO DOUBLE-WORD
         SPACE  1
EDISECTL EQU    (*-EDISECT)/8  LENGTH OF 'EDISECT' IN D.W.
         SPACE  3
*    BITS DEFINED IN 'EDIFLG1'
         SPACE  1
EDITEXEC EQU    X'80'          EDITOR INVOKED FROM WITHIN 'EXEC'
EDITNMSG EQU    X'40'          EDITOR INVOKED WITH '&CONTROL NOMSG'
EDITDCSS EQU    X'20'          EDITOR RUNNING IN DCSS
```

LICENSED MATERIAL - PROPERTY OF IBM

LVLSECT

EDIT RECURSION LEVEL CONTROL BLOCK

```
LVLSECT  DSECT  ,          LAYOUT OF 'LEVEL' CONTROL BLOCK AND DATA AREA
LVLF     DS     1F               FORWARD POINTER TO NEXT 'LVLSECT'
LVLB     DS     1F               BACKWARD POINTER TO NEXT 'LVLSECT'
LVLSCR   DS     CL1920           CURRENT 'SCREEN' AT THIS LEVEL
LVLTAB   DS     XL240            TAB TABLE FOR THIS LEVEL
LVLFLDT  DS     70H              TABLE FOR UP TO 70 FIELDS ON SCREEN
LVLFLDTL EQU    *-2              LAST AVAILABLE POSITION (ALLOW 69 FIELDS)
LVLRPT   DS     1F               "REPEAT" REC IMAGE
LVLCRSR  DS     1H               CURSOR ADDRESS ON SCREEN
LVLSW1   DS     1X               GENERAL PURPOSE EDIT LEVEL STATS
LVLSW2   DS     1X               EDIT LEVEL OPTION STATS
LVLSW3   DS     1X               TYPE I (LEVEL DISPOSITION) STATS
LVLSW4   DS     1X               MORE SECOND LEVEL OPTION STATS
LVLTID   DS     CL18             TEMPORARY LEVEL FILEID BUCKET
         DS     1F               RESERVED
LVLCPRI  DS     1X               CURRENT CURSOR PRIORITY
LVLSERCH DS     CL3              SERIALIZATION CHARACTERS TO BE USED...
LVLSERIN DS     1F               SERIALIZATION INCREMENT VALUE
LVLFID   DS     OCL18            FILE ID AT THIS RECURSION LEVEL
LVLFN    DS     CL8              FILE NAME FOR THIS EDITOR LEVEL
LVLFT    DS     CL8              FILE TYPE FOR THIS EDITOR LEVEL
LVLFM    DS     CL2              FILE MODE FOR THIS EDITOR LEVEL
LVLRECL  DS     1H               LOGICAL RECORD LENGTH FOR THIS LEVEL
LVLCNT   DS     1F               COUNT OF ITEMS CURRENTLY IN FILE
LVLSPARE DS     1F               POINTER TO START OF SPARE DATA BLOCKS
LVLSTORE DS     1F               PTR. TO FIRST STORAGE BLOCK
LVLTOF   DS     1F               PTR. TO TOP-OF-FILE
LVLEOF   DS     1F               PTR. TO END-OF-FILE
LVLLDB   DS     1F               PTR. TO LOGICAL DISPLAY BLOCK CHAIN
LVLPT    DS     1F               ANCHOR TO 'POINT' CHAIN..
LVLDUMMY DS     1F               PTR. TO DUMMY DATA BLOCK
LVLCHG   DS     1F               PTR. TO LEVEL CHANGE TABLE
LVLAUTCT DS     1H               'AUTOSAVE' CURRENT COUNT
LVLAUTST DS     1H               'AUTOSAVE' CURRENT SETTING
LVLAUTID DS     CL8              'AUTOSAVE' IDENTIFIER
LVLDEFEX DS     OCL8             NAME OF DEFAULT EXEC TO BE CALLED
LVLFMTHW DS     12H              'FORMAT' SPEC. HALF-WORDS
         CNOP   0,8              ALIGN END TO DOUBLE-WORD
LVLSECTL EQU    (*-LVLSECT)/8    LENGTH OF BLOCK IN DOUBLE-WORDS
```

LICENSED MATERIAL - PROPERTY OF IBM

```
*   BITS DEFINED IN 'LVLSW1'

LVLERRM  EQU   X'80'          INDICATES ERROR MESSAGE ON SCREEN
LVLFSCR  EQU   X'40'          FULL-SCREEN MUST BE WRITTEN
LVLPQ    EQU   X'20'          LEVEL IS QUEUED FOR PROCESSING
LVLCNTCH EQU   X'10'          INDICATES COUNT HAS CHANGED
LVLNEW   EQU   X'08'          LEVEL IS CREATING A "NEW" FILE
LVLDQ    EQU   X'04'          LEVEL IS QUEUED FOR DISPLAY
LVLPCSR  EQU   X'01'          LEVEL CONTAINS PRIMARY CURSOR


*   BITS DEFINED IN 'LVLSW2'

LVLVARI  EQU   X'80'          INDICATES RECFM=V
LVLDBLNK EQU   X'40'          'DBLANK' OPTION REQUESTED
LVLNULL  EQU   X'20'          'NULLS' OPTION SET
LVLNUMB  EQU   X'10'          'NUMBERS' OPTION SET
LVLUPD   EQU   X'08'          'UPDATE' OPTION SET
LVLSEQ8  EQU   X'04'          'SEQ8' OPTION SET
LVLCTL   EQU   X'02'          'CTL' OPTION INDICATED
LVLNORD  EQU   X'01'          'NOREAD' OPTION SET


*   BITS DEFINED IN 'LVLSW3'

LVLQUIT  EQU   X'80'          LEVEL TO BE DISCARDED
LVLSAVE  EQU   X'40'          LEVEL TO BE STORED
LVLFMT   EQU   X'20'          LEVEL TO BE FORMATTED
LVLGLOB  EQU   X'10'          PERFORM QUIT/SAVE ON ALL OTHER LEVELS
LVLDISP  EQU   LVLQUIT+LVLSAVE+LVLGLOB


*   BITS DEFINED IN 'LVLSW4'

LVLCNVT  EQU   X'80'          'CONVERT' OPTION SET


*   DEFINED CURSOR PRIORITIES (IN 'LVLCPRI')

CPRINULL EQU   X'00'          NO CURSOR POSITION YET ESTABLISHED
CPRIDEF  EQU   X'01'          CURSOR IN DEFAULT LOCATION
CPRIPDEF EQU   X'02'          CURSOR IN PRIMARY DEFAULT LOCATION
CPRIT1   EQU   X'03'          CURSOR PRIORITY FOR TYPE I CMND LINE
CPRIT2   EQU   X'04'          CURSOR PRIORITY FOR TYPE II CMND LINE
CPRIADD  EQU   X'05'          CURSOR PRIORITY FOR 'ADDED' LINE
CPRIINP  EQU   X'06'          CURSOR PRIORITY FOR 'INPUT' DDB
```

LICENSED MATERIAL - PROPERTY OF IBM

LDBSECT

LOGICAL DISPLAY BLOCK FORMAT

```
LDBSECT   DSECT ,              LAYOUT OF LOGICAL DISPLAY BLOCK
LDBFWD    DS    1F             PTR. TO NEXT LOGICAL DISPLAY BLOCK
LDBACK    DS    1F             PTR. TO PREVIOUS LOGICAL DISPLAY BLOCK
LDBLPTR   DS    1F             LOGICAL LINE PTR. FOR THIS DISPLAY
LDBREC    DS    22F            'REC' PTR. TABLE FOR DATA AREAS
LDBRECN   EQU   *              END OF 'LDBREC' (MAX. 22 FIELDS)
LDBTABS   DS    26H            TABLE OF TABS FOR THIS DISPLAY (MAX. 25)
LDBTABSN  EQU   *              END OF 'LDBTABS' TABLE
LDBVIEW   DS    26F            TABLE OF 'VIEW' PTRS. (MAX. 25 FLDS)
LDBVIEWN  EQU   *              END OF 'LDBVIEW'
LDBDDBW   DS    1H             WIDTH OF EACH DATA DISPLAY BUFFER
LDBADR    DS    1H             STARTING BUFFER ADDRESS FOR THIS DISPLAY
LDBZONE1  DS    1H             BEGIN ZONE FOR 'CHANGE', 'LOCATE'
LDBZONE2  DS    1H             END ZONE FOR 'CHANGE', 'LOCATE'
LDBVOFF   DS    1H             'VIEW' OFFSET VALUE
LDBSW1    DS    1X             BANK OF GENERAL PURPOSE STATS...
LDBSW2    DS    1X             ...FOR THIS DISPLAY
LDBINPTR  DS    1F             "INPUT POINTER"
LDBADDPT  DS    AL3            "ADD" LINE POINTER
          DS    XL1            RESERVED
LDBPFTB   DS    1F             POINTER TO PF TABLE FOR THIS DISPLAY
LDBCHRIN  DS    CL8            "INPUT" CONSTANT
LDBCMD    DS    CL132          LOGICAL DISPLAY COMMAND LINE AREA
LDBRETCD  DS    1F             RETURN CODE FROM LAST COMMAND
          CNOP  0,8            ASSURE DOUBLE-WORD ALIGNMENT

LDBSECTL  EQU   (*-LDBSECT)/8  LENGTH OF 'LDBSECT' (IN D.W.)


    BITS DEFINED IN 'LDBSW1'

LDBS1T3   EQU   X'80'          INDICATES THIS DISPLAY HAS TYPE III AREAS
LDBSINSD  EQU   X'40'          NON-SEQUENTIAL DISPLAY (SET-UP BY COMMAND
LDBCHTAB  EQU   X'20'          RESET TAB BITS FOR THIS DISPLAY
LDBUC     EQU   X'10'          TRANS. INPUT TO UPPER CASE
LDBINPUT  EQU   X'08'          DISPLAY IS IN "INPUT" MODE
LDBCMMSG  EQU   X'04'          INDICATES ON-SCREEN CMND. LINE HAS MSG.
LDBPRC    EQU   X'02'          PROCESSING COMPLETE ON THIS LDB
LDBCSRP   EQU   X'01'          THIS DISPLAY HAS THE CURSOR


*   BITS DEFINED IN 'LDBSW2'

LDBCMDH   EQU   X'80'          'COMMAND HOLD' IN EFFECT
```

LICENSED MATERIAL - PROPERTY OF IBM

```
*   BITS DEFINED IN BYTE 0 OF EACH 'LDBREC' ENTRY

LDREC80  EQU   X'80'          NO 'REC' ENTRY (END OF TABLE)
LDRECCST EQU   X'40'          'REC' PTR. IS TO A 'CONSTANT'
LDRECINP EQU   X'20'          DDB IS FOR 'INPUT' (NO 'REC' EXISTS)


*   'LDBVIEW' ENTRY DEFINITION

LDVWDS   DSECT  ,
LDVWFLAG DS    1H             BITS 0-4 ARE UNSED AS FLAGS
LDVWSTRT EQU   X'0FFF'        REMAINING BITS ARE STAT DISPLACEMENT
LDVWLNG  DS    1H             STD./HEX VIEW FIELD LENGTH
                              (LOGICAL TABLE END IF LENGTH .LT. 1)
LDVW     EQU   LDVWDS,*-LDVWDS 'LDBVIEW' TABLE ENTRY


*   BITS DEFINED IN 'LDVWFLAG'

LDVWSTD  EQU   X'F0'          IF ZERO, STANDARD VIEWED FIELD
LDVWHEX  EQU   X'80'          INDICATE COLUMNS TO BE VIEWED IN HEX
LDVWSTAT EQU   X'F000'
```

LICENSED MATERIAL - PROPERTY OF IBM

SAVEAREA

STANDARD SAVE AREA FOR 3270 EDITOR

```
SAVEAREA DSECT ,                  STANDARD REGISTER SAVE AREA
SAVER0   DS    1F                 SAVE AREA FOR CALLER'S R0
SAVER1   DS    1F                 SAVE AREA FOR CALLER'S R1
SAVER2   DS    1F                 SAVE AREA FOR CALLER'S R2
SAVER3   DS    1F                 SAVE AREA FOR CALLER'S R3
SAVER4   DS    1F                 SAVE AREA FOR CALLER'S R4
SAVER5   DS    1F                 SAVE AREA FOR CALLER'S R5
SAVER6   DS    1F                 SAVE AREA FOR CALLER'S R6
SAVER7   DS    1F                 SAVE AREA FOR CALLER'S R7
SAVER8   DS    1F                 SAVE AREA FOR CALLER'S R8
SAVER9   DS    1F                 SAVE AREA FOR CALLER'S R9
SAVER10  DS    1F                 SAVE AREA FOR CALLER'S R10
SAVER11  DS    1F                 SAVE AREA FOR CALLER'S R11
SAVER12  DS    1F                 SAVE AREA FOR CALLER'S R12
SAVER13  DS    1F                 SAVE AREA FOR CALLER'S R13
SAVER14  DS    1F                 SAVE AREA FOR CALLER'S R14
SAVER15  DS    1F                 SAVE AREA FOR CALLER'S R15
SAVEPNT  DS    1F                 PTR. TO NEXT SAVE AREA IN 'FREE' CHAIN
SAVERET  EQU   SAVEPNT            RETURN ADDR TO 'EDICALL' WHEN S.A. IN USE
SAVWRK1  DS    1F                 MAY BE USED BY CALLED ROUTINE
SAVWRK2  DS    1F                 MAY BE USED BY CALLED ROUTINE
SAVWRK3  DS    1F                 MAY BE USED BY CALLED ROUTINE

SAVEAREL EQU   (*-SAVEAREA)/8     NUMBER OF DOUBLE-WORDS IN SAVE AREA
```

LICENSED MATERIAL - PROPERTY OF IBM

IOSECT -

LAYOUT OF CONTROL BLOCK USED BY I/O MANAGER

```
IOSECT    DSECT  ,              LAYOUT OF I/O CONTROL BLOCK
IOCCWS    DS     0XL16          START OF CONSOLE CHAN PGM. AREA
IOPLIST   DS     1D             P-LIST FOR CONSOLE I/O
IOCONBUF  DC     AL1(1),AL3(0)  CONSOLE BUFFER ADDR (EDIBUFF)
IOCODE    DS     0CL2           CONSOLE I/O EDIT CODES
IOCODE1   DS     CL1            CONSOLE I/O EDIT TYPE CODE A
IOCODE2   DS     CL1            CONSOLE I/O EDIT TYPE CODE B
IOCONCNT  DS     1H             BUCKET FOR I/O DATA LENGTH
IOCRSR    DS     1H             CURRENT I/O CURSOR LOCATION
IOAID     DS     1X             ATTENTION CHARACTER RECEIVED
IOSW1     DS     1X             CURRENT I/O STATUS INDICATOR
IOSTAX    DS     5F             STAX PARM. LIST BUILD AREA
IOSCRW    DS     1H             WIDTH OF THE 3270 SCREEN
IOSCRL    DS     1H             LENGTH OF THE 3270 SCREEN
IOSCRSZ   DS     1H             SIZE OF THE 3270 SCREEN
IOCSRSTK  DS     5H             CURSOR STACK
IOCSRDPT  EQU    5              MAX. DEPTH OF CURSOR STACK
IOPF1     DS     1F             'PF1' ASSIGNMENT ANCHOR WORD
IOPF2     DS     1F             'PF2' ASSIGNMENT ANCHOR WORD
IOPF3     DS     1F             'PF3' ASSIGNMENT ANCHOR WORD
IOPF4     DS     1F             'PF4' ASSIGNMENT ANCHOR WORD
IOPF5     DS     1F             'PF5' ASSIGNMENT ANCHOR WORD
IOPF6     DS     1F             'PF6' ASSIGNMENT ANCHOR WORD
IOPF7     DS     1F             'PF7' ASSIGNMENT ANCHOR WORD
IOPF8     DS     1F             'PF8' ASSIGNMENT ANCHOR WORD
IOPF9     DS     1F             'PF9' ASSIGNMENT ANCHOR WORD
IOPF10    DS     1F             'PF10' ASSIGNMENT ANCHOR WORD
IOPF11    DS     1F             'PF11' ASSIGNMENT ANCHOR WORD
IOPF12    DS     1F             'PF12' ASSIGNMENT ANCHOR WORD
IOTAB     DS     XL240          REAL SCREEN TAB TABLE
IOFLDT    DS     256H           REAL SCREEN FIELD TABLE
IOCMDLT   DS     14H            ADDRESSES OF SCREEN COMMAND LINES
IOSW2     DS     1X             SECOND BANK OF I/O CONTROL STATS
IOTABC    DS     1C             SPECIFIED TAB CHARACTER
          DS     1H             RESERVED
IOLPAD    DS     1F             LOGICAL TABBING PAD CHARACTER DEFIN.
          SVCBOOT ,             GENERATE SVC INTERFACE ROUTINE
          DS     0D             ALIGN IMAGE I/O BUFFER TO D.W.
IOSCREEN  DS     CL(L'LVLSCR)   SCREEN IMAGE I/O BUFFER
          DS     0D             ALIGN CONTROL BLOCK LENGTH TO D.W.

IOSECTL   EQU    (*-IOSECT)/8   LENGTH OF 'IOSECT' IN DOUBLE-WORDS


*         BITS DEFINED IN 'IOSW1'
```

LICENSED MATERIAL - PROPERTY OF IBM

```
IOCLRSC  EQU   X'80'          SIGNAL ISSUE "CLEAR" SCREEN BEFORE WRITE
IOFSCR   EQU   X'40'          INDICATES FULL-SCREEN I/O REQUIRED
IOALRM   EQU   X'20'          INDICATES 3270 ALARM TO BE SET
IOERCONT EQU   X'0C'          ERROR=CONTINUE IN EFFECT IF ZEROS
IOERESCP EQU   X'08'          ERROR=ESCAPE IN EFFECT
IOERVER  EQU   X'04'          ERROR=VERIFY IN EFFECT
IOCOMP   EQU   X'02'          INDICATES I/O COMPLETE

*        BITS DEFINED IN 'IOSW2'

IOCMPRS  EQU   X'80'          INDICATES COMPRESS OPTION IN EFFECT
IOAPL    EQU   X'40'          I/O TO BE PERFORMED IN APL MODE
IOTEXT   EQU   X'20'          I/O TO BE PERFORMED IN TEXT MODE
IOISRT   EQU   X'10'          PHYS. VIRT. SCREEN IN INSERT MODE
```

LICENSED MATERIAL - PROPERTY OF IBM

REC

LAYOUT OF IN-STORAGE RECORDS

```
REC       DSECT ,              START OF FILE RECORD IN STORAGE
RECFWD    DS    1F             PTR. TO NEXT RECORD IN STORAGE
RECBACK   DS    1F             PTR. TO PREVIOUS IN STORAGE
RECDATA   DS    0C             START OF RECORD DATA AREA (LENGTH
                               DEPENDENT ON 'LVLRECL')
```

STORE

FORMAT OF 'STORE' BLOCKS FOR DATA

```
STORE     DSECT ,              START OF VIRTUAL STORAGE BLOCK
STOREF    DS    1F             PTR. TO NEXT BLOCK OF STORAGE
STORES    DS    1F             LENGTH OF THIS STORAGE BLOCK (IN D.W.)
STORER1   DS    0D             START OF FIRST 'REC' IN THIS BLOCK
```

PTSECT

NAMED POINT DIRECTORY BLOCK FORMAT

```
PTSECT    DSECT  ,             LAYOUT OF NAMED POINT DIRECTORY BLOCK
PTNEXT    DS    1F             FORWARD PTR. TO NEXT BLOCK IN DIRECTORY
PTLAST    DS    1F             ADDRESS OF "LAST" ACTIVE ENTRY IN BLOCK
PTENTRY   DS    0XL8           START OF INDIVIDUAL DIRECTORY ENTRY
PTSW      DS    XL1            ENTRY OPTION CONTROL STATS
PTREC     DS    AL3            'REC' ADDRESS ASSIGNED TO THIS ENTRY
PTNAME    DS    CL4            ENTRY NAME (1-4 CHARACTERS)
PTSECTL   EQU   AUXBLKS*8      LENGTH OF ONE POINT DIRECTORY BLOCK

*    BITS DEFINED IN 'PTSW'

PTLOCK    EQU   X'80'          INDICATES THIS IS A "LOCKED" NAME
```

LICENSED MATERIAL - PROPERTY OF IBM

PROSECT

FORMAT OF FILETYPE PROFILE CONTROL BLOCK

```
PROSECT  DSECT  ,              FILETYPE PROFILE CONTROL BLOCK FORMAT
PROPTR   DS     1F             FORWARD CONTROL BLOCK CHAIN
PROLRECL DS     1F             PROFILE'D LOGICAL RECD. LENGTH
PROFT    DS     CL8            PROFILE'D FILETYPE
PROFM    DS     CL2            PROFILE'D FILEMODE
PROPT    DS     1X             PROFILE'D LEVEL OPTIONS
PROABBR  DS     1X             MINIMUM ABBREVIATION LENGTH
         DS     1F             RESERVED
PROCTL   DS     CL8            PROFILE'D CONTROL FILENAME
PRODEF   DS     CL8            PROFILE'D DEFAULTS EXEC NAME
PROSYN   DS     CL8            PROFILE'D COMMAND SYNONYM
PROSECTL EQU    (*-PROSECT)/8  LENGTH OF PROFILE CONTROL BLOCK

*   BITS DEFINED IN 'PROPT'

PROUPD   EQU    X'80'          SIGNAL UPDATE OPTION PROTOTYPE'D
PROSEQ8  EQU    X'40'          SIGNAL 'SEQ8' OPTION PROTOTYPE'D
```

LICENSED MATERIAL - PROPERTY OF IBM

CRT1, CRT2, AND CRT3

LAYOUT OF TYPE I, II, AND III COMMAND AREAS ON SCREEN

```
CRT1      DSECT  ,              FORMAT OF FILE INFORMATION LINE
CRT1X1    DS     1X             ATTRIBUTE BYTE FOR FILE ID FIELD
CRT1FID   DS     0CL20          FILE IDENTIFICATION AREA
CRT1FN    DS     CL8            FILE NAME
          DS     CL1            FILLER
CRT1FT    DS     CL8            FILE TYPE
          DS     CL1            FILLER
CRT1FM    DS     CL2            FILE MODE
CRT1X2    DS     1X             ATTRIBUTE BYTE FOR PROTECTED STATUS FIELD
CRT1STAT  DS     0CL58          CURRENT STATUS INFORMATION
CRT1NEWF  DS     C              NEW-FILE SIGNAL ('-' OR '*')
          DS     C              FILLER
CRT1CHG   DS     CL3            "CHANGE" LEVEL INDICATOR
          DS     CL13           RESERVED
CRT1RECF  DS     CL7            RECORD FORMAT ('F' OR 'V' AND LRECL)
          DS     C              FILLER
CRT1CST4  DS     C              CONSTANT - '('
CRT1AUTO  DS     CL8            'AUTOSAVE' IDENTIFIER
CRT1CST3  DS     C              CONSTANT - ')'
          DS     C              FILLER
CRT1CST2  DS     CL4            CONSTANT - 'ALT='
CRT1ALT   DS     CL5            CURRENT FILE ALTERATION COUNT
          DS     C              FILLER
CRT1CST1  DS     CL5            CONSTANT - 'RECDS'
          DS     C              FILLER
CRT1CNT   DS     CL5            COUNT OF ACTIVE RECORDS
CRT1LDB   DS     0X             START OF ATTRIBUTE BYTE FOR FIRST LDB

CRT1L     EQU    *-CRT1         LENGTH OF FILE INFORMATION LNE
```

LICENSED MATERIAL - PROPERTY OF IBM

```
CRT2      DSECT ,                 LAYOUT OF LOGICAL DISPLAY COMMAND AREA
CRT2X1    DS    1X                ATTRIBUTE BYTE FOR COMMAND LINE POINTER
CRT2PTR   DS    CL4               '===>' FOR COMMAND AREA
CRT2X2    DS    1X                ATTRIBUTE BYTE FOR COMMAND AREA
CRT2CMD   DS    CL73              TYPE I/TYPE II COMMAND LINE
CRT2X3    DS    1X                ATTRIBUTE BYTE FOR FIRST DDB
CRT2DDB   DS    0C                FIRST DATA DISPLAY BUFFER

CRT2L     EQU   *-CRT2            LENGTH OF LOG. DISPLAY COMMAND AREA


CRT3      DSECT ,                 FORMAT OF TYPE III COMMAND AREA
CRT3X1    DS    1X                ATTRIBUTE BYTE TO START COMMAND AREA
CRT3CMD   DS    CL6               TYPE III COMMAND AREA
CRT3X2    DS    1X                ATTRIBUTE BYTE ENDING TYPE III CMND AREA

CRT3L     EQU   *-CRT3            LENGTH OF TYPE III COMMAND AREA
```

LICENSED MATERIAL - PROPERTY OF IBM

GENERAL EDITOR EQUATES

SCREEN AND FIELD TABLE BIT DEFINITIONS

```
FLDCHG    EQU    X'80'          ATTRIBUTE - FIELD ALTERED BY EDITOR
FLDMOD    EQU    X'40'          ATTRIBUTE - FIELD ALTERED BY USER
FLDNULL   EQU    X'02'          CONVERT TRAILING FIELD BLANKS TO NULLS
```

GENERAL EDITOR INTERNAL EQUATES

```
AUXBLKS   EQU    168/8          LENGTH OF AUXILLARY STORAGE (IN D.W.)
MAXSIZE   EQU    2047           MAXIMUM ALLOWABLE FILE SIZE
MAXITEM   EQU    65533          MAXIMUM SIZE OF DISK-RESIDENT FILE
TTLSAVE   EQU    15             NUMBER OF INTERNAL SAVE AREAS....
```

LICENSED MATERIAL - PROPERTY OF IBM

ECOMMAND IS A CMS PROBLEM PROGRAM WHICH EXECUTES IN THE TRANSIENT AREA AND USES THE EDGAR SVC INTERFACE. ECOMMAND TAKES IT'S INPUT EITHER IN THE FORM OF ENTRY PARAMETERS ENTERED ON THE SAME LINE, OR FROM THE CONSOLE. IF THE INPUT IS ENTERED AS PARAMETERS AND THE USER WISHES TO AVOID THE TOKENIZING NORMALLY DONE BY CMS, HE MAY "WRAP" THE DATA WHICH HE WISHES TO REMAIN CONTIGUOUS WITH PARENTHESES AND ENTER IT IN THE NORMAL EIGHT (8) BYTE GROUPINGS. FOR EXAMPLE

ECOMMAND L (/CONFID ENTIAL/)

THE POSSIBLE RETURN CODES WHICH CAN BE RETURNED FROM ECOMMAND ARE:

- 0 - INDICATING THE COMMAND EXECUTED SUCCESSFULLY
- -1 - INDICATING THE COMMAND EXECUTED SUCCESSFULLY AND PRODUCED A MESSAGE WHICH WOULD NORMALLY BE PLACED ON THE COMMAND LINE OF THE SCREEN. AND HAS BEEN PLACED LIFO IN THE CMS CONSOLE STACK.
- 1 - INDICATING THAT THE DISPLAY EDITING SYSTEM IS NOT ACTIVE
- XXX - A MESSAGE NUMBER CORRESPONDING TO AN EDGAR ERROR MESSAGE INDICATING THE UNSUCESSFUL EXECUTION OF A COMMAND. THE ERROR MESSAGE IS ALSO PLACE LIFO IN THE CMS CONSOLE STACK IF THE COMMAND WAS CALLED VIA SVC.

LICENSED MATERIAL - PROPERTY OF IBM

LY20-2311-0

LICENSED MATERIAL – PROPERTY OF IBM

Display Editing System for CMS System Guide Printed in U.S.A. LY20-2311-0

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, N.Y. 10604

IBM World Trade Americas/Far East Corporation
Town of Mount Pleasant, Route 9, North Tarrytown, N.Y., U.S.A. 10591

IBM World Trade Europe/Middle East/Africa Corporation
360 Hamilton Avenue, White Plains, N.Y., U.S.A. 10601

LICENSED MATERIAL – PROPERTY OF IBM

Display Editing System for CMS
System Guide

LY20-2311-0

READER'S COMMENT FORM

This manual is part of a library that serves as a reference source for systems analysts, programmers, and operators of IBM systems. This form may be used to communicate your views about this publication. They will be sent to the author's department for whatever review and action, if any, is deemed appropriate. Comments may be written in your own language; use of English is not required.

IBM shall have the nonexclusive right, in its discretion, to use and distribute all submitted information, in any form, for any and all purposes, without obligation of any kind to the submitter. Your interest is appreciated.

Note: *Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.*

Possible topics for comment are:

Clarity Accuracy Completeness Organization Coding Retrieval Legibility

If you wish a reply, give your name and mailing address:

_____

_____

_____

Note: Staples can cause problems with automated mail sorting equipment. Please use pressure sensitive or other gummed tape to seal this form.

What is your occupation? _____

Number of latest Newsletter associated with this publication: _____

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A. (Elsewhere, an IBM

Reader's Comment Form

LICENSED MATERIAL – PROPERTY OF IBM

Fold and tape | Please Do Not Staple | Fold and tape

First Class
Permit 40
Armonk
New York

Business Reply Mail
No postage stamp necessary if mailed in the U.S.A.

Postage will be paid by:

International Business Machines Corporation
Department 825
1133 Westchester Avenue
White Plains, New York 10604

Fold and tape | Please Do Not Staple | Fold and tape

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, N.Y. 10604

IBM World Trade Americas/Far East Corporation
Town of Mount Pleasant, Route 9, North Tarrytown, N.Y., U.S.A. 10591

IBM World Trade Europe/Middle East/Africa Corporation
360 Hamilton Avenue, White Plains, N.Y., U.S.A. 10601

Cut or Fold Along Line

LICENSED MATERIAL – PROPERTY OF IBM

Display Editing System for CMS
System Guide

LY20-2311-0

READER'S COMMENT FORM

This manual is part of a library that serves as a reference source for systems analysts, programmers, and operators of IBM systems. This form may be used to communicate your views about this publication. They will be sent to the author's department for whatever review and action, if any, is deemed appropriate. Comments may be written in your own language; use of English is not required.

IBM shall have the nonexclusive right, in its discretion, to use and distribute all submitted information, in any form, for any and all purposes, without obligation of any kind to the submitter. Your interest is appreciated.

Note: *Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.*

Possible topics for comment are:

Clarity Accuracy Completeness Organization Coding Retrieval Legibility

If you wish a reply, give your name and mailing address:

_____

_____

_____

Note: Staples can cause problems with automated mail sorting equipment. Please use pressure sensitive or other gummed tape to seal this form.

What is your occupation? _____

Number of latest Newsletter associated with this publication: _____

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A. (Elsewhere, an IBM office or representative will be happy to forward your comments or you may mail directly to the address

Reader's Comment Form

LICENSED MATERIAL – PROPERTY OF IBM

Fold and tape | Please Do Not Staple | Fold and tape

First Class
Permit 40
Armonk
New York

Business Reply Mail

No postage stamp necessary if mailed in the U.S.A.

Postage will be paid by:

International Business Machines Corporation
Department 825
1133 Westchester Avenue
White Plains, New York 10604

Fold and tape | Please Do Not Staple | Fold and tape

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, N.Y. 10604

IBM World Trade Americas/Far East Corporation
Town of Mount Pleasant, Route 9, North Tarrytown, N.Y., U.S.A. 10591

IBM World Trade Europe/Middle East/Africa Corporation
360 Hamilton Avenue, White Plains, N.Y., U.S.A. 10601